

VDS RPC API

H.W. Hesselink, Voipro

July 2, 2011

Contents

1	Adminperson functions	7
1.1	Add Adminperson	7
1.2	Update Adminperson	8
1.3	Delete Adminperson	8
1.4	List Adminperson	8
1.5	Show Adminperson	8
1.6	Showm Adminperson	8
2	Agent functions	9
2.1	Add Agent	9
2.2	Update Agent	9
2.3	Delete Agent	9
2.4	List Agent	9
2.5	Show Agent	9
2.6	Showm Agent	9
3	Callroute functions	10
3.1	Add Callroute	10
3.2	Update Callroute	10
3.3	Delete Callroute	10
3.4	List Callroute	10
3.5	Show Callroute	10
3.6	Showm Callroute	11
4	Callroutetype functions	12
4.1	Add Callroutetype	12
4.2	Update Callroutetype	12
4.3	Delete Callroutetype	12
4.4	List Callroutetype	12
4.5	Show Callroutetype	12

4.6	Showm Callroutetype	12
5	Carrierlisttype functions	13
5.1	Add Carrierlisttype	13
5.2	Update Carrierlisttype	13
5.3	Delete Carrierlisttype	13
5.4	List Carrierlisttype	13
5.5	Show Carrierlisttype	13
5.6	Showm Carrierlisttype	13
6	Channeltype functions	14
6.1	Add Channeltype	14
6.2	Update Channeltype	14
6.3	Delete Channeltype	14
6.4	List Channeltype	14
6.5	Show Channeltype	14
6.6	Showm Channeltype	14
7	Choiceslot functions	15
7.1	Add Choiceslot	15
7.2	Update Choiceslot	15
7.3	Delete Choiceslot	15
7.4	List Choiceslot	15
7.5	Show Choiceslot	15
7.6	Showm Choiceslot	15
8	Company functions	16
8.1	Add Company	16
8.2	Update Company	17
8.3	Delete Company	17
8.4	List Company	17
8.5	Show Company	17
8.6	Showm Company	18
9	Contactperson functions	19
9.1	Add Contactperson	19
9.2	Update Contactperson	19
9.3	Delete Contactperson	19
9.4	List Contactperson	19
9.5	Show Contactperson	19
9.6	Showm Contactperson	19

10 Dateslot functions	20
10.1 Add Dateslot	20
10.2 Update Dateslot	20
10.3 Delete Dateslot	20
10.4 List Dateslot	20
10.5 Show Dateslot	20
10.6 Showm Dateslot	20
11 Dayofweek functions	21
11.1 Add Dayofweek	21
11.2 Update Dayofweek	21
11.3 Delete Dayofweek	21
11.4 List Dayofweek	21
11.5 Show Dayofweek	21
11.6 Showm Dayofweek	21
12 Did functions	22
12.1 Add Did	22
12.2 Update Did	22
12.3 Delete Did	22
12.4 List Did	22
12.5 Show Did	22
12.6 Showm Did	23
13 Extnumber functions	24
13.1 Add Extnumber	24
13.2 Update Extnumber	24
13.3 Delete Extnumber	24
13.4 List Extnumber	24
13.5 Show Extnumber	24
13.6 Showm Extnumber	24
14 Huntgroup functions	25
14.1 Add Huntgroup	25
14.2 Update Huntgroup	25
14.3 Delete Huntgroup	26
14.4 List Huntgroup	26
14.5 Show Huntgroup	26
14.6 Showm Huntgroup	26
15 Huntgrouprule functions	27

15.1 Add Huntgrouprule	27
15.2 Update Huntgrouprule	27
15.3 Delete Huntgrouprule	27
15.4 List Huntgrouprule	27
15.5 Show Huntgrouprule	27
15.6 Showm Huntgrouprule	27
16 Ipallow functions	28
16.1 Add Ipallow	28
16.2 Update Ipallow	28
16.3 Delete Ipallow	28
16.4 List Ipallow	28
16.5 Show Ipallow	28
16.6 Showm Ipallow	28
17 Language functions	29
17.1 Add Language	29
17.2 Update Language	29
17.3 Delete Language	29
17.4 List Language	29
17.5 Show Language	29
17.6 Showm Language	29
18 Person functions	30
18.1 Add Person	30
18.2 Update Person	31
18.3 Delete Person	31
18.4 List Person	32
18.5 Show Person	32
18.6 Showm Person	32
19 Phone functions	33
19.1 Add Phone	33
19.2 Update Phone	33
19.3 Delete Phone	33
19.4 List Phone	34
19.5 Show Phone	34
19.6 Showm Phone	34
20 Phonebook functions	35
20.1 Add Phonebook	35

20.2 Update Phonebook	35
20.3 Delete Phonebook	35
20.4 List Phonebook	35
20.5 Show Phonebook	35
20.6 Showm Phonebook	36
21 Phonenumber functions	37
21.1 Add Phonenumber	37
21.2 Update Phonenumber	37
21.3 Delete Phonenumber	37
21.4 List Phonenumber	37
21.5 Show Phonenumber	37
21.6 Showm Phonenumber	38
22 Phonetype functions	39
22.1 Add Phonetype	39
22.2 Update Phonetype	39
22.3 Delete Phonetype	39
22.4 List Phonetype	39
22.5 Show Phonetype	40
22.6 Showm Phonetype	40
23 Pickupgroup functions	41
23.1 Add Pickupgroup	41
23.2 Update Pickupgroup	41
23.3 Delete Pickupgroup	41
23.4 List Pickupgroup	41
23.5 Show Pickupgroup	41
23.6 Showm Pickupgroup	41
24 Queue functions	42
24.1 Add Queue	42
24.2 Update Queue	43
24.3 Delete Queue	43
24.4 List Queue	43
24.5 Show Queue	43
24.6 Showm Queue	43
25 Queue_member functions	44
25.1 Add Queue_member	44
25.2 Delete Queue_member	44

26 Server functions	45
26.1 Add Server	45
26.2 Update Server	45
26.3 Delete Server	45
26.4 List Server	45
26.5 Show Server	45
26.6 Showm Server	45
27 Siptrunk functions	46
27.1 Add Siptrunk	46
27.2 Update Siptrunk	46
27.3 Delete Siptrunk	46
27.4 List Siptrunk	46
27.5 Show Siptrunk	46
27.6 Showm Siptrunk	46
28 Supplier functions	47
28.1 Add Supplier	47
28.2 Update Supplier	47
28.3 Delete Supplier	47
28.4 List Supplier	47
28.5 Show Supplier	47
28.6 Showm Supplier	48
29 Timeslot functions	49
29.1 Add Timeslot	49
29.2 Update Timeslot	49
29.3 Delete Timeslot	49
29.4 List Timeslot	49
29.5 Show Timeslot	49
29.6 Showm Timeslot	49
30 Trunk functions	50
30.1 Add Trunk	50
30.2 Update Trunk	50
30.3 Delete Trunk	50
30.4 List Trunk	51
30.5 Show Trunk	51
30.6 Showm Trunk	51
31 Vmtimezone functions	52

31.1 Add Vmtimezone	52
31.2 Update Vmtimezone	52
31.3 Delete Vmtimezone	52
31.4 List Vmtimezone	52
31.5 Show Vmtimezone	52
31.6 Showm Vmtimezone	52
32 Miscellaneous functions	53
32.1 Agentpause	53
32.2 Agentunpause	53
32.3 Ast Reload	53
32.4 Ast Update	53
32.5 Autologin Agent	53
32.6 Autologout Agent	53
32.7 Calc Call Cost	53
32.8 Finger	53
32.9 List Cdrs	53
32.10List Codecs	54
32.11List Company Directory	54
32.12Login	54
32.13Logout	54
32.14Makecall	54
32.15Pause Queue Member	54
32.16Set Cdr Cost	54
32.17Set Loglevel	54
32.18Show Company Directory	55
32.19Show Loggedin Details	55
32.20Show Loggedin Names	55
32.21Unpause Queue Member	55

1 Adminperson functions

1.1 Add Adminperson

Synopsis add_adminperson(username, adminpassword, personid, *optargs)

Required arguments • adminpassword - text

- personid - integer references person(id)
- username - text

Optional arguments • emailaddr - text

- restricted - boolean

1.2 Update Adminperson

Synopsis update_adminperson(key, fields, policy=None)

Required arguments • key - integer

- fields - a dict (Python) or hash (Perl) of fields to update, see the help for the add_adminperson call for valid fields.

1.3 Delete Adminperson

Synopsis delete_adminperson(key, policy=None)

Required arguments • key - integer

1.4 List Adminperson

Synopsis list_adminperson(*q)

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the add_adminperson call for valid condition names.

1.5 Show Adminperson

Synopsis show_adminperson(key, policy=None)

Required arguments • key - integer

1.6 Showm Adminperson

Synopsis showm_adminperson(*q)

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the add_adminperson call for valid condition names.

2 Agent functions

2.1 Add Agent

Synopsis `add_agent(queueid, personid, *optargs)`

Required arguments

- `personid` - integer references `person(id)`
- `queueid` - integer references `queue(id)`

Optional arguments

- `autologin` - boolean

2.2 Update Agent

Synopsis `update_agent(key, fields, policy=None)`

Required arguments

- `key` - integer
- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_agent` call for valid fields.

2.3 Delete Agent

Synopsis `delete_agent(key, policy=None)`

Required arguments

- `key` - integer

2.4 List Agent

Synopsis `list_agent(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_agent` call for valid condition names.

2.5 Show Agent

Synopsis `show_agent(key, policy=None)`

Required arguments

- `key` - integer

2.6 Showm Agent

Synopsis `showm_agent(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the `showm`, see the help for the `add_agent` call for valid condition names.

3 Callroute functions

3.1 Add Callroute

Synopsis `add_callroute(type_id, compid, name, description, *optargs)`

Required arguments • `compid` - integer references company(id)

- `description` - text
- `name` - text
- `type_id` - integer references callroute_type(id)

Optional arguments • `data` - text

- `fail_exten` - integer
- `fail_route` - integer references callroute(id)
- `next_exten` - integer
- `next_route` - integer references callroute(id)

3.2 Update Callroute

Synopsis `update_callroute(key, fields, policy=None)`

Required arguments • `key` - integer

- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_callroute` call for valid fields.

3.3 Delete Callroute

Synopsis `delete_callroute(key, policy=None)`

Required arguments • `key` - integer

3.4 List Callroute

Synopsis `list_callroute(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_callroute` call for valid condition names.

3.5 Show Callroute

Synopsis `show_callroute(key, policy=None)`

Required arguments • `key` - integer

3.6 Showm Callroute

Synopsis `showm_callroute(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_callroute` call for valid condition names.

4 Callroutetype functions

4.1 Add Callroutetype

Synopsis `add_callroutetype(name, *optargs)`

Required arguments • name - text

Optional arguments

4.2 Update Callroutetype

Synopsis `update_callroutetype(key, fields, policy=None)`

Required arguments • key - integer

- fields - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_callroutetype` call for valid fields.

4.3 Delete Callroutetype

Synopsis `delete_callroutetype(key, policy=None)`

Required arguments • key - integer

4.4 List Callroutetype

Synopsis `list_callroutetype(*q)`

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_callroutetype` call for valid condition names.

4.5 Show Callroutetype

Synopsis `show_callroutetype(key, policy=None)`

Required arguments • key - integer

4.6 Showm Callroutetype

Synopsis `showm_callroutetype(*q)`

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_callroutetype` call for valid condition names.

5 Carrierlisttype functions

5.1 Add Carrierlisttype

Synopsis `add_carrierlisttype(name, description)`

Required arguments • name - text

Optional arguments • description - text

5.2 Update Carrierlisttype

Synopsis `update_carrierlisttype(key, fields, policy=None)`

Required arguments • key - integer

- fields - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_carrierlisttype` call for valid fields.

5.3 Delete Carrierlisttype

Synopsis `delete_carrierlisttype(key, policy=None)`

Required arguments • key - integer

5.4 List Carrierlisttype

Synopsis `list_carrierlisttype(*q)`

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_carrierlisttype` call for valid condition names.

5.5 Show Carrierlisttype

Synopsis `show_carrierlisttype(key, policy=None)`

Required arguments • key - integer

5.6 Showm Carrierlisttype

Synopsis `showm_carrierlisttype(*q)`

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_carrierlisttype` call for valid condition names.

6 Channeltype functions

6.1 Add Channeltype

Synopsis `add_channeltype(type)`

Required arguments • `type` - text

Optional arguments

6.2 Update Channeltype

Synopsis `update_channeltype(key, fields, policy=None)`

Required arguments • `key` - integer

- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_channeltype` call for valid fields.

6.3 Delete Channeltype

Synopsis `delete_channeltype(key, policy=None)`

Required arguments • `key` - integer

6.4 List Channeltype

Synopsis `list_channeltype(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_channeltype` call for valid condition names.

6.5 Show Channeltype

Synopsis `show_channeltype(key, policy=None)`

Required arguments • `key` - integer

6.6 Showm Channeltype

Synopsis `showm_channeltype(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_channeltype` call for valid condition names.

7 Choiceslot functions

7.1 Add Choiceslot

Synopsis `add_choiceslot(callroute_id, choice, *optargs)`

Required arguments

- `callroute_id` - integer references `callroute(id)`
- `choice` - integer

Optional arguments

- `next_exten` - integer
- `next_route` - integer references `callroute(id)`

7.2 Update Choiceslot

Synopsis `update_choiceslot(key, fields, policy=None)`

Required arguments

- `key` - integer
- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_choiceslot` call for valid fields.

7.3 Delete Choiceslot

Synopsis `delete_choiceslot(key, policy=None)`

Required arguments

- `key` - integer

7.4 List Choiceslot

Synopsis `list_choiceslot(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_choiceslot` call for valid condition names.

7.5 Show Choiceslot

Synopsis `show_choiceslot(key, policy=None)`

Required arguments

- `key` - integer

7.6 Showm Choiceslot

Synopsis `showm_choiceslot(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_choiceslot` call for valid condition names.

8 Company functions

8.1 Add Company

Synopsis add_company(shortname, name, countrycode, areacode, extnlength, linelength, langid, timezoneid, trunkout, *optargs)

Required arguments • areacode - integer

- countrycode - integer
- extnlength - integer
- langid - integer references language(id)
- linelength - integer
- name - text
- shortname - text
- timezoneid - integer references voicemail_tzmsgs(id)
- trunkout - integer references trunk(id)

Optional arguments • address - text

- address2 - text
- allowg729 - boolean
- allowroaming - boolean
- budget - numeric
- budget_max_pct - integer
- budgeted - boolean
- callerid - integer references did(id)
- carrier_list_id - integer references carrier_list(id)
- checklogout - boolean
- country - text
- email - text
- enabled - boolean
- fax - text
- fwdorigclid - boolean
- monitor - boolean
- msrvr1 - integer references server(id)
- msrvr2 - integer references server(id)
- operpanel - boolean
- phone - text
- postcode - text
- proxy1 - integer references server(id)
- proxy2 - integer references server(id)

- reference_code - text
- server - integer references server(id)
- showdnid - boolean
- town - text
- utstariff - boolean

Description Register a company (customer) and activate services. The add_company call adds a company to the OnDemand PBX system, with a billing address and contact information.

8.2 Update Company

Synopsis update_company(key, fields, policy=None)

Required arguments • key - integer

- fields - a dict (Python) or hash (Perl) of fields to update, see the help for the add_company call for valid fields.

Description Update company information. This call updates all information if a certain customer with the new information.

8.3 Delete Company

Synopsis delete_company(key, policy=None)

Required arguments • key - integer

Description Delete a company from the system. The delete_company call disables a company in the OnDemand PBX system. Historical information will be kept as necessary. No phones or persons may be associated with the company for the call to succeed.

8.4 List Company

Synopsis list_company(*q)

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the add_company call for valid condition names.

8.5 Show Company

Synopsis show_company(key, policy=None)

Required arguments • key - integer

Description Show information about a company. Returns all information about the named company.

8.6 Showm Company

Synopsis `showm_company(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_company` call for valid condition names.

9 Contactperson functions

9.1 Add Contactperson

Synopsis `add_contactperson(lastname, supplierid, *optargs)`

Required arguments

- `lastname` - text
- `supplierid` - integer references `supplier(id)`

Optional arguments

- `email` - text
- `fax` - text
- `firstname` - text
- `middlename` - text
- `mobile` - text
- `phone` - text

9.2 Update Contactperson

Synopsis `update_contactperson(key, fields, policy=None)`

Required arguments

- `key` - integer
- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_contactperson` call for valid fields.

9.3 Delete Contactperson

Synopsis `delete_contactperson(key, policy=None)`

Required arguments

- `key` - integer

9.4 List Contactperson

Synopsis `list_contactperson(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_contactperson` call for valid condition names.

9.5 Show Contactperson

Synopsis `show_contactperson(key, policy=None)`

Required arguments

- `key` - integer

9.6 Showm Contactperson

Synopsis `showm_contactperson(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_contactperson` call for valid condition names.

10 DATESLOT FUNCTIONS

10.1 Add Dateslot

Synopsis `add_dateslot(callroute_id, every_year, slot_start, slot_stop, *optargs)`

Required arguments

- `callroute_id` - integer references `callroute(id)`

- `slot_start` - timestampz
- `slot_stop` - timestampz

Optional arguments

- `every_year` - boolean

10.2 Update Dateslot

Synopsis `update_dateslot(key, fields, policy=None)`

Required arguments

- `key` - integer

- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_dateslot` call for valid fields.

10.3 Delete Dateslot

Synopsis `delete_dateslot(key, policy=None)`

Required arguments

- `key` - integer

10.4 List Dateslot

Synopsis `list_dateslot(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_dateslot` call for valid condition names.

10.5 Show Dateslot

Synopsis `show_dateslot(key, policy=None)`

Required arguments

- `key` - integer

10.6 Showm Dateslot

Synopsis `showm_dateslot(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_dateslot` call for valid condition names.

11 Dayofweek functions

11.1 Add Dayofweek

Synopsis `add_dayofweek(name, *optargs)`

Required arguments • name - bpchar

Optional arguments

11.2 Update Dayofweek

Synopsis `update_dayofweek(key, fields, policy=None)`

Required arguments • key - integer

- fields - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_dayofweek` call for valid fields.

11.3 Delete Dayofweek

Synopsis `delete_dayofweek(key, policy=None)`

Required arguments • key - integer

11.4 List Dayofweek

Synopsis `list_dayofweek(*q)`

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_dayofweek` call for valid condition names.

11.5 Show Dayofweek

Synopsis `show_dayofweek(key, policy=None)`

Required arguments • key - integer

11.6 Showm Dayofweek

Synopsis `showm_dayofweek(*q)`

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_dayofweek` call for valid condition names.

12 Did functions

12.1 Add Did

Synopsis `add_did(external_nr_id, compid, number, *optargs)`

Required arguments

- `compid` - integer references company(id)
- `external_nr_id` - integer references external_number(id)
- `number` - integer

Optional arguments

- `timeout` - integer

Description Add DID (Direct Inward Dialing) number. The `add_did` call adds a DID, a Direct Inward Dialing number, to a company's dialplan. The external number is mapped onto either a phone or internal group.

12.2 Update Did

Synopsis `update_did(key, fields, policy=None)`

Required arguments

- `key` - integer
- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_did` call for valid fields.

Description Update DID configuration. Updates the configuration of the given DID with the new information given.

12.3 Delete Did

Synopsis `delete_did(key, policy=None)`

Required arguments

- `key` - integer

Description Delete DID. Removes DID from it's dialplan.

12.4 List Did

Synopsis `list_did(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_did` call for valid condition names.

12.5 Show Did

Synopsis `show_did(key, policy=None)`

Required arguments

- `key` - integer

Description Show information about a DID. Returns all information about a given DID.

12.6 Showm Did

Synopsis `showm_did(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_did` call for valid condition names.

13 Extnumber functions

13.1 Add Extnumber

Synopsis `add_extnumber(country, areacode, subscriber, trunkid, *optargs)`

Required arguments • areacode - integer

- country - integer
- subscriber - integer
- trunkid - integer references `trunk(id)`

Optional arguments • inuse - boolean

13.2 Update Extnumber

Synopsis `update_extnumber(key, fields, policy=None)`

Required arguments • key - integer

- fields - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_extnumber` call for valid fields.

13.3 Delete Extnumber

Synopsis `delete_extnumber(key, policy=None)`

Required arguments • key - integer

13.4 List Extnumber

Synopsis `list_extnumber(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_extnumber` call for valid condition names.

13.5 Show Extnumber

Synopsis `show_extnumber(key, policy=None)`

Required arguments • key - integer

13.6 Showm Extnumber

Synopsis `showm_extnumber(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the `showm`, see the help for the `add_extnumber` call for valid condition names.

14 Huntgroup functions

14.1 Add Huntgroup

Synopsis `add_huntgroup(compid, name, groupid, fullname, langid, timezoneid, *optargs)`

Required arguments • `compid` - integer references `company(id)`

- `fullname` - text
- `groupid` - integer
- `langid` - integer references `language(id)`
- `name` - text
- `timezoneid` - integer references `voicemail_tzmsgs(id)`

Optional arguments • `cyclical` - boolean

- `dovm` - boolean
- `fallbacknr` - text
- `isqueue` - boolean
- `looping` - boolean
- `password` - integer
- `showindirectory` - boolean
- `vmannounceonly` - boolean
- `vmattach` - boolean
- `vmdelete` - boolean
- `vmemailaddr` - text
- `vmemailnotify` - boolean
- `vminstructions` - boolean

Description Add hunt group to a company's dialplan. The `add_huntgroup` call adds a hunt group to the dialplan of the specified company. A hunt group consists of zero or more phones which all ring when the group is dialed, until the first phone answers (or a timeout occurs).

14.2 Update Huntgroup

Synopsis `update_huntgroup(key, fields, policy=None)`

Required arguments • `key` - integer

- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_huntgroup` call for valid fields.

Description Update hunt group. Updates the configuration of a certain hunt group with the information provided.

14.3 Delete Huntgroup

Synopsis delete_huntgroup(key, policy=None)

Required arguments • key - integer

Description Delete hunt group from company's dialplan. Removes a hunt group from a company's dialplan. If any DIDs map onto this hunt group, the call fails.

14.4 List Huntgroup

Synopsis list_huntgroup(*q)

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the add_huntgroup call for valid condition names.

14.5 Show Huntgroup

Synopsis show_huntgroup(key, policy=None)

Required arguments • key - integer

Description Show information about a hunt group. Returns all information about a certain hunt group.

14.6 Showm Huntgroup

Synopsis showm_huntgroup(*q)

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the add_huntgroup call for valid condition names.

15 Huntgroup rule functions

15.1 Add Huntgroup rule

Synopsis `add_huntgroup_rule(huntgroup_id, ruleorder, phones, *optargs)`

Required arguments • `huntgroup_id` - integer references `huntgroup(id)`

- `phones` - text
- `ruleorder` - integer

Optional arguments • `timeout` - integer

15.2 Update Huntgroup rule

Synopsis `update_huntgroup_rule(key, fields, policy=None)`

Required arguments • `key` - integer

- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_huntgroup_rule` call for valid fields.

15.3 Delete Huntgroup rule

Synopsis `delete_huntgroup_rule(key, policy=None)`

Required arguments • `key` - integer

15.4 List Huntgroup rule

Synopsis `list_huntgroup_rule(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_huntgroup_rule` call for valid condition names.

15.5 Show Huntgroup rule

Synopsis `show_huntgroup_rule(key, policy=None)`

Required arguments • `key` - integer

15.6 Showm Huntgroup rule

Synopsis `showm_huntgroup_rule(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_huntgroup_rule` call for valid condition names.

16 Ipallow functions

16.1 Add Ipallow

Synopsis `add_ipallow(compid, inetaddr)`

Required arguments

- `compid` - integer references company(id)
- `inetaddr` - text

Optional arguments

- `description` - text
- `expires` - timestampz

16.2 Update Ipallow

Synopsis `update_ipallow(key, fields, policy=None)`

Required arguments

- `key` - integer
- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_ipallow` call for valid fields.

16.3 Delete Ipallow

Synopsis `delete_ipallow(key, policy=None)`

Required arguments

- `key` - integer

16.4 List Ipallow

Synopsis `list_ipallow(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_ipallow` call for valid condition names.

16.5 Show Ipallow

Synopsis `show_ipallow(key, policy=None)`

Required arguments

- `key` - integer

16.6 Showm Ipallow

Synopsis `showm_ipallow(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the `showm`, see the help for the `add_ipallow` call for valid condition names.

17 Language functions

17.1 Add Language

Synopsis `add_language(iso2letter, name, localname)`

Required arguments • `iso2letter` - text

- `localname` - text
- `name` - text

Optional arguments

17.2 Update Language

Synopsis `update_language(key, fields, policy=None)`

Required arguments • `key` - integer

- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_language` call for valid fields.

17.3 Delete Language

Synopsis `delete_language(key, policy=None)`

Required arguments • `key` - integer

17.4 List Language

Synopsis `list_language(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_language` call for valid condition names.

17.5 Show Language

Synopsis `show_language(key, policy=None)`

Required arguments • `key` - integer

17.6 Showm Language

Synopsis `showm_language(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_language` call for valid condition names.

18 Person functions

18.1 Add Person

Synopsis add_person(userid, lastname, compid, langid, timezoneid, *optargs)

Required arguments • compid - integer references company(id)

- langid - integer references language(id)
- lastname - text
- timezoneid - integer references voicemail_tzmsgs(id)
- userid - integer

Optional arguments • budget - numeric

- budgeted - boolean
- busynr - text
- callerid - integer references did(id)
- callwaiting - boolean
- conferencepin - integer
- cp_adult - boolean
- cp_international - boolean
- cp_local - boolean
- cp_longdistance - boolean
- cp_mobile - boolean
- cp_premium - boolean
- cp_tollfree - boolean
- dndnr - text
- donotdisturb - boolean
- fax2email - boolean
- firstname - text
- fwdorigclid - boolean
- handlebusy - boolean
- handleloggedout - boolean
- handlenoanswer - boolean
- handleunavail - boolean
- home_phone - text
- incoming2mobile - boolean
- isconference - boolean
- isfaxdata - boolean
- location - text
- lockedphoneid - integer references phone(id)

- loggedoutnr - text
- middlename - text
- mobile_phone - text
- noanswernr - text
- noanswertimeout - integer
- password - integer
- phonenumber - integer references phonenumber(id)
- showindirectory - boolean
- unavailnr - text
- vmannounceonly - boolean
- vmattach - boolean
- vmdelete - boolean
- vmemailaddr - text
- vmemailnotify - boolean
- vmenabled - boolean
- vmifbusy - boolean
- vmifdnd - boolean
- vmifloggedout - boolean
- vmifnoanswer - boolean
- vmifunavail - boolean
- vminstructions - boolean

Description Add person to company. This call adds a person to a company's directory. A person has a name, location information, username and password, and a voicemailbox with pin. A person must be created before a phone can be associated with him/her and he/she can actually be called.

18.2 Update Person

Synopsis update_person(key, fields, policy=None)

Required arguments • key - integer

- fields - a dict (Python) or hash (Perl) of fields to update, see the help for the add_person call for valid fields.

Description Update information about a person. This call updates all information about a certain person. Care must be taken with this call.

18.3 Delete Person

Synopsis delete_person(key, policy=None)

Required arguments • key - integer

Description Delete person from company directory. Deletes the person from the company's directory. If person is associated with a phone, that association is first broken. Relevant historical information may be kept for legal or business purposes at all times.

18.4 List Person

Synopsis `list_person(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_person` call for valid condition names.

18.5 Show Person

Synopsis `show_person(key, policy=None)`

Required arguments • `key` - integer

Description Show information about a certain person. Returns all known information about a certain person.

18.6 Showm Person

Synopsis `showm_person(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the shown, see the help for the `add_person` call for valid condition names.

19 Phone functions

19.1 Add Phone

Synopsis `add_phone(compid, name, phonetypeid, channeltype, macaddr, adminpassword, *optargs)`

Required arguments

- `adminpassword` - text
- `channeltype` - integer references `channeltype(id)`
- `compid` - integer references `company(id)`
- `macaddr` - macaddr
- `name` - text
- `phonetypeid` - integer references `phonetype(id)`

Optional arguments

- `allowroaming` - boolean
- `autocomplete` - boolean
- `call_waiting` - boolean
- `description` - text
- `enabled` - boolean
- `firmware` - text
- `inetaddr` - text
- `location` - text
- `maxlines` - integer

Description Add phone with given characteristics and associate it with a certain user. The `add_phone` call registers a phone to the OnDemand PBX system. The phone is registered to a company, and phone characteristics such as type, protocol used, location are initialized. The initial/default user is associated with the phone.

19.2 Update Phone

Synopsis `update_phone(key, fields, policy=None)`

Required arguments

- `key` - integer
- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_phone` call for valid fields.

Description Update phone information. Updates all phone information in the OnDemand PBX system with the given information.

19.3 Delete Phone

Synopsis `delete_phone(key, policy=None)`

Required arguments

- `key` - integer

Description Delete phone from system. Delete a phone from the OnDemand PBX system. This action is irreversible. Billing information and other logging is preserved.

19.4 List Phone

Synopsis `list_phone(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_phone` call for valid condition names.

19.5 Show Phone

Synopsis `show_phone(key, policy=None)`

Required arguments • `key` - integer

Description Return information of indicated phone. Returns all information about the phone.

19.6 Showm Phone

Synopsis `showm_phone(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_phone` call for valid condition names.

20 Phonebook functions

20.1 Add Phonebook

Synopsis `add_phonebook(compid, persid, lastname, *optargs)`

Required arguments • `compid` - integer references company(id)

- `lastname` - text
- `persid` - integer references person(id)

Optional arguments • `firstname` - text

- `homenr` - text
- `middlename` - text
- `mobilenr` - text
- `othernr` - text
- `worknr` - text

20.2 Update Phonebook

Synopsis `update_phonebook(key, fields, policy=None)`

Required arguments • `key` - integer

- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_phonebook` call for valid fields.

20.3 Delete Phonebook

Synopsis `delete_phonebook(key, policy=None)`

Required arguments • `key` - integer

20.4 List Phonebook

Synopsis `list_phonebook(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_phonebook` call for valid condition names.

20.5 Show Phonebook

Synopsis `show_phonebook(key, policy=None)`

Required arguments • `key` - integer

20.6 Showm Phonebook

Synopsis `showm_phonebook(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_phonebook` call for valid condition names.

21 Phonenumber functions

21.1 Add Phonenumber

Synopsis `add_phonenumber(phone_id, line, number, username, secret, *optargs)`

Required arguments

- `number` - integer
- `phone_id` - integer references `phone(id)`
- `secret` - text
- `username` - text

Optional arguments

- `cancallout` - boolean
- `canreceive` - boolean
- `channel` - text
- `description` - text
- `line` - integer
- `maxcalls` - integer
- `proxyport` - integer
- `proxysrvr` - text
- `status` - text

21.2 Update Phonenumber

Synopsis `update_phonenumber(key, fields, policy=None)`

Required arguments

- `key` - integer
- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_phonenumber` call for valid fields.

21.3 Delete Phonenumber

Synopsis `delete_phonenumber(key, policy=None)`

Required arguments

- `key` - integer

21.4 List Phonenumber

Synopsis `list_phonenumber(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_phonenumber` call for valid condition names.

21.5 Show Phonenumber

Synopsis `show_phonenumber(key, policy=None)`

Required arguments

- `key` - integer

21.6 Showm Phoneline

Synopsis `showm_phoneline(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_phoneline` call for valid condition names.

22 Phonetype functions

22.1 Add Phonetype

Synopsis `add_phonetype(name, shortname, *optargs)`

Required arguments

- `name` - text
- `shortname` - text

Optional arguments

- `dtmfmode` - text
- `has_g729` - boolean
- `manufacturer` - text
- `model` - text
- `numlines` - integer
- `showinselect` - boolean
- `showorder` - integer
- `supported` - boolean

Description Add type of phone. The `add_phonetype` call adds a phone type to the OnDemand PBX system, and sets certain characteristics used e.g. for provisioning.

22.2 Update Phonetype

Synopsis `update_phonetype(key, fields, policy=None)`

Required arguments

- `key` - integer
- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_phonetype` call for valid fields.

Description Update information about certain phone type. The `update_phonetype` call updates all known information about a certain type of phone (brand, model etc).

22.3 Delete Phonetype

Synopsis `delete_phonetype(key, policy=None)`

Required arguments

- `key` - integer

Description Delete phone type. Deletes the indicated phone type from the system. Only possible if no registered phones exist of this type.

22.4 List Phonetype

Synopsis `list_phonetype(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_phonetype` call for valid condition names.

22.5 Show Phonetype

Synopsis `show_phonetype(key, policy=None)`

Required arguments • `key` - integer

Description Show information about a certain phone type. Returns all known information about a certain type of phone.

22.6 Showm Phonetype

Synopsis `showm_phonetype(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_phonetype` call for valid condition names.

23 Pickupgroup functions

23.1 Add Pickupgroup

Synopsis `add_pickupgroup(compid, name, userids)`

Required arguments • `compid` - integer references company(id)

- `name` - text
- `userids` - text

Optional arguments

23.2 Update Pickupgroup

Synopsis `update_pickupgroup(key, fields, policy=None)`

Required arguments • `key` - integer

- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_pickupgroup` call for valid fields.

23.3 Delete Pickupgroup

Synopsis `delete_pickupgroup(key, policy=None)`

Required arguments • `key` - integer

23.4 List Pickupgroup

Synopsis `list_pickupgroup(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_pickupgroup` call for valid condition names.

23.5 Show Pickupgroup

Synopsis `show_pickupgroup(key, policy=None)`

Required arguments • `key` - integer

23.6 Showm Pickupgroup

Synopsis `showm_pickupgroup(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_pickupgroup` call for valid condition names.

24 Queue functions

24.1 Add Queue

Synopsis `add_queue(fullname, compid, groupid, userids, *optargs)`

Required arguments • `compid` - integer references company(id)

- `fullname` - text
- `groupid` - integer
- `name` - text
- `userids` - text

Optional arguments • `announce` - text

- `announce_frequency` - integer
- `announce_holdtime` - text
- `announce_round_seconds` - integer
- `context` - text
- `eventmemberstatus` - boolean
- `eventwhencalled` - boolean
- `joinempty` - text
- `leavewhenempty` - text
- `maxlen` - integer
- `memberdelay` - integer
- `monitor_format` - text
- `monitor_join` - boolean
- `musiconhold` - text
- `operpanel` - boolean
- `periodic_announce` - text
- `periodic_announce_frequency` - integer
- `queue_callswaiting` - text
- `queue_holdtime` - text
- `queue_lessthan` - text
- `queue_minutes` - text
- `queue_reporthold` - text
- `queue_seconds` - text
- `queue_thankyou` - text
- `queue_thereare` - text
- `queue_youarenext` - text
- `reportholdtime` - boolean
- `retry` - integer

- servicelevel - integer
- shortname - text
- strategy - text
- timeout - integer
- timeoutrestart - boolean
- weight - integer
- wrapuptime - integer

24.2 Update Queue

Synopsis `update_queue(key, fields, policy=None)`

Required arguments • key - integer

- fields - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_queue` call for valid fields.

24.3 Delete Queue

Synopsis `delete_queue(key, policy=None)`

Required arguments • key - integer

24.4 List Queue

Synopsis `list_queue(*q)`

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_queue` call for valid condition names.

24.5 Show Queue

Synopsis `show_queue(key, policy=None)`

Required arguments • key - integer

24.6 Showm Queue

Synopsis `showm_queue(*q)`

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_queue` call for valid condition names.

25 Queue_member functions

25.1 Add Queue_member

Synopsis `add_queue_member(queueid, personid)`

25.2 Delete Queue_member

Synopsis `delete_queue_member(queueid, personid)`

26 Server functions

26.1 Add Server

Synopsis `add_server(inetaddr, secret, *optargs)`

Required arguments • `inetaddr` - text

- `internaladdr` - text
- `secret` - text

Optional arguments • `description` - text

- `has_g729` - boolean
- `mfid` - integer references `records(id)`
- `name` - text

26.2 Update Server

Synopsis `update_server(key, fields, policy=None)`

Required arguments • `key` - integer

- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_server` call for valid fields.

26.3 Delete Server

Synopsis `delete_server(key, policy=None)`

Required arguments • `key` - integer

26.4 List Server

Synopsis `list_server(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_server` call for valid condition names.

26.5 Show Server

Synopsis `show_server(key, policy=None)`

Required arguments • `key` - integer

26.6 Showm Server

Synopsis `showm_server(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the `showm`, see the help for the `add_server` call for valid condition names.

27 Siptrunk functions

27.1 Add Siptrunk

Synopsis `add_siptrunk(compid, siptrunkid, name, description)`

Required arguments • `compid` - integer references company(id)

- `description` - text
- `name` - text
- `siptrunkid` - integer

Optional arguments

27.2 Update Siptrunk

Synopsis `update_siptrunk(key, fields, policy=None)`

Required arguments • `key` - integer

- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_siptrunk` call for valid fields.

27.3 Delete Siptrunk

Synopsis `delete_siptrunk(key, policy=None)`

Required arguments • `key` - integer

27.4 List Siptrunk

Synopsis `list_siptrunk(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_siptrunk` call for valid condition names.

27.5 Show Siptrunk

Synopsis `show_siptrunk(key, policy=None)`

Required arguments • `key` - integer

27.6 Showm Siptrunk

Synopsis `showm_siptrunk(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_siptrunk` call for valid condition names.

28 Supplier functions

28.1 Add Supplier

Synopsis `add_supplier(shortname, name, *optargs)`

Required arguments

- name - text
- shortname - text

Optional arguments

- address - text
- address2 - text
- country - text
- email - text
- fax - text
- phone - text
- postcode - text
- town - text

28.2 Update Supplier

Synopsis `update_supplier(key, fields, policy=None)`

Required arguments

- key - integer
- fields - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_supplier` call for valid fields.

28.3 Delete Supplier

Synopsis `delete_supplier(key, policy=None)`

Required arguments

- key - integer

28.4 List Supplier

Synopsis `list_supplier(*q)`

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_supplier` call for valid condition names.

28.5 Show Supplier

Synopsis `show_supplier(key, policy=None)`

Required arguments

- key - integer

28.6 Showm Supplier

Synopsis `showm_supplier(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_supplier` call for valid condition names.

29 Timeslot functions

29.1 Add Timeslot

Synopsis `add_timeslot(callroute_id, day_of_week, slot_start, slot_stop, *optargs)`

Required arguments • `callroute_id` - integer references `callroute(id)`

- `day_of_week` - integer references `day_of_week(id)`
- `slot_start` - time
- `slot_stop` - time

Optional arguments

29.2 Update Timeslot

Synopsis `update_timeslot(key, fields, policy=None)`

Required arguments • `key` - integer

- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_timeslot` call for valid fields.

29.3 Delete Timeslot

Synopsis `delete_timeslot(key, policy=None)`

Required arguments • `key` - integer

29.4 List Timeslot

Synopsis `list_timeslot(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_timeslot` call for valid condition names.

29.5 Show Timeslot

Synopsis `show_timeslot(key, policy=None)`

Required arguments • `key` - integer

29.6 Showm Timeslot

Synopsis `showm_timeslot(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the `showm`, see the help for the `add_timeslot` call for valid condition names.

30 Trunk functions

30.1 Add Trunk

Synopsis `add_trunk(name, channel, channeltype, *optargs)`

Required arguments

- `channel` - text
- `channeltype` - integer references `channeltype(id)`
- `name` - text

Optional arguments

- `calllimit` - integer
- `dropleadingzero` - boolean
- `dtmfmode` - text
- `host` - text
- `incominghascc` - boolean
- `nrprefix` - text
- `options` - text
- `port` - integer
- `server` - integer references `server(id)`
- `supplierid` - integer references `supplier(id)`

Description Add trunk. Add a trunk with given name to the OnDemand PBX system. Currently, only PRI trunks can be added this way.

30.2 Update Trunk

Synopsis `update_trunk(key, fields, policy=None)`

Required arguments

- `key` - integer
- `fields` - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_trunk` call for valid fields.

Description Update trunk. The `update_trunk` call updates the information about the named trunk.

30.3 Delete Trunk

Synopsis `delete_trunk(key, policy=None)`

Required arguments

- `key` - integer

Description Delete trunk. Delete a trunk from the system. Only possible if no external numbers are connected with this trunk.

30.4 List Trunk

Synopsis `list_trunk(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_trunk` call for valid condition names.

30.5 Show Trunk

Synopsis `show_trunk(key, policy=None)`

Required arguments • `key` - integer

Description Show information about a trunk. Returns all information about a named trunk.

30.6 Showm Trunk

Synopsis `showm_trunk(*q)`

Optional arguments `q` - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_trunk` call for valid condition names.

31 Vmtimezone functions

31.1 Add Vmtimezone

Synopsis `add_vmtimezone(name, timezone, message)`

Required arguments • message - text

- name - text
- timezone - text

Optional arguments

31.2 Update Vmtimezone

Synopsis `update_vmtimezone(key, fields, policy=None)`

Required arguments • key - integer

- fields - a dict (Python) or hash (Perl) of fields to update, see the help for the `add_vmtimezone` call for valid fields.

31.3 Delete Vmtimezone

Synopsis `delete_vmtimezone(key, policy=None)`

Required arguments • key - integer

31.4 List Vmtimezone

Synopsis `list_vmtimezone(*q)`

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the list, see the help for the `add_vmtimezone` call for valid condition names.

31.5 Show Vmtimezone

Synopsis `show_vmtimezone(key, policy=None)`

Required arguments • key - integer

31.6 Showm Vmtimezone

Synopsis `showm_vmtimezone(*q)`

Optional arguments q - a dict (Python) or hash (Perl) of conditions limiting the showm, see the help for the `add_vmtimezone` call for valid condition names.

32 Miscellaneous functions

32.1 Agentpause

Synopsis agentpause(compid, line, queuenr=None)

32.2 Agentunpause

Synopsis agentunpause(compid, line, queuenr=None)

32.3 Ast Reload

Synopsis ast_reload(mods, from_update=False)

Description Reload one or more PBX modules, all modules if 'mods' is empty. If mods is not empty it must be either a single module name, or a list of modules, from firewall, voicemail, extensions, iax.

32.4 Ast Update

Synopsis ast_update(mods)

Description Write config file(s) and reload the matching PBX module(s), all configs and modules if 'mods' is empty. If mods is not empty it must be either a single module name, or a list of modules, from conference, voicemail, extensions, iax.

32.5 Autologin Agent

Synopsis autologin_agent(personid)

32.6 Autologout Agent

Synopsis autologout_agent(personid)

32.7 Calc Call Cost

Synopsis calc_call_cost(company, destnr, date, duration)

32.8 Finger

Synopsis finger()

Description Return a list of currently logged-in users.

32.9 List Cdrs

Synopsis listcdrs(*optargs)

Description Return a list of Call Detail Records containing the following fields: calldate, clid, dst, billsec, disposition, cost Optargs is an optional dict (Python) or hash (Perl) defining one or more of: company - only show records for this company userid - only show records (incoming and outgoing) for this user sortorder - one or more of the fields listed above as an SQL ORDER BY clause, e.g. “dst desc, billsec” would sort first on dst in descending order and then on billsec in ascending order maxdays - only show records from up to maxdays ago.

32.10 List Codecs

Synopsis list_codecs()

32.11 List Company Directory

Synopsis list_company_directory(compid, *sortorder)

Description List people in company compid where showindirectory is true. The optional sortorder is a comma-separated list of fields to sort on.

32.12 Login

Synopsis login(userid, company, number)

Description Log the person (userid, company) in to phonenumber.

32.13 Logout

Synopsis logout(userid, company, number)

Description Log person (userid, company) off from phonenumber.

32.14 Makecall

32.15 Pause Queue Member

Synopsis pause_queue_member(queueid, personid)

32.16 Set Cdr Cost

Synopsis set_cdr_cost(uniqueid)

32.17 Set Loglevel

32.18 Show Company Directory

Synopsis `show_company_directory(id)`

32.19 Show Loggedin Details

Synopsis `show_loggedin_details(phoneid, line)`

Description Return a list of currently logged-in users.

32.20 Show Loggedin Names

Synopsis `show_loggedin_names(phoneid)`

Description Return a list of currently logged-in users.

32.21 Unpause Queue Member

Synopsis `unpause_queue_member(queueid, personid)`